



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/893,829	06/28/2001	Raja Krishnaswamy	MS174293.01 (MSFTP243US)	5228
27195 7590 07/17/2009 TUROC & WATSON, LLP 127 Public Square 57th Floor, Key Tower CLEVELAND, OH 44114			EXAMINER CHANKONG, DOHM	
			ART UNIT	PAPER NUMBER
			2452	
			NOTIFICATION DATE	DELIVERY MODE
			07/17/2009 ELECTRONIC	

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

docket1@thepatentattorneys.com  
hholmes@thepatentattorneys.com  
lpasterchek@thepatentattorneys.com

<b>Office Action Summary</b>	<b>Application No.</b> 09/893,829	<b>Applicant(s)</b> KRISHNASWAMY ET AL.	
	<b>Examiner</b> DOHM CHANKONG	<b>Art Unit</b> 2452	

**-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --**

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).

Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

1) ☒ Responsive to communication(s) filed on 27 April 2009.

2a) ☒ This action is **FINAL**.                      2b) ☐ This action is non-final.

3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

4) ☒ Claim(s) 1-8, 10-15, 17-23 and 27 is/are pending in the application.

    4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.

5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.

6) ☒ Claim(s) 1-8, 10-15, 17-23 and 27 is/are rejected.

7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.

8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

9) ☐ The specification is objected to by the Examiner.

10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a) ☐ All    b) ☐ Some \*    c) ☐ None of:

1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

1) ☐ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO/SB/08)  
    Paper No(s)/Mail Date 3/11/2009

4) ☐ Interview Summary (PTO-413)  
    Paper No(s)/Mail Date \_\_\_\_\_

5) ☐ Notice of Informal Patent Application

6) ☐ Other: \_\_\_\_\_

### **DETAILED ACTION**

1. This action is in response to Applicant's amendment filed on 4/27/2009. Claims 1, 6, 10, 14, 15, 17, 19, 23, and 27 are amended. Claims 9, 16, and 24-26 were previously cancelled. Accordingly, claims 1-8, 10-15, 17-23, and 27 are presented for further examination.

2. This action is a final rejection.

### ***Response to Arguments***

3. Applicant's arguments with respect to claims 1-8, 10-15, 17-23, and 27 have been considered but are moot in view of the new ground(s) of rejection.

### ***Claim Rejections - 35 USC § 103***

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

4. The rejection of the claim limitation directed to the interceptor's accessibility to application code is informed by the Applicant's specification. Specifically, Applicant's specification states that "intercepting a method call and making such interception accessible to an application developer can include receiving control and receiving a data structure...populated with information concerning the intercepted method call" [Applicant's specification, pg. 3, lines 1-4]. Further, the information concerning the call can include method names and input parameter

information [pg. 3, lines 4-9]. For the purposes of this rejection, the interpretation of an "interceptor accessible to application code" relies on this explanation within Applicant's specification.

5. Claims 1-8 and 10-14 are rejected under 35 U.S.C §103(a) as being unpatentable over Arnold et al, U.S Patent No. 6.393.497 ["Arnold"], in view of Hollander et al, U.S Patent No. 6.823.460 ["Hollander"], in further view of Applicant's admitted prior art [citing Application's publication 20050268309].

6. All citations are to Arnold unless otherwise noted.

7. As to claim 1, Arnold discloses a system for interacting with an object, the system comprising:

at least one processor that executes the following computer executable components stored on at least one computer readable medium:

a method call interceptor that intercepts a method call to an object [*Hollander*, column 4 «lines 33-39»] and that directly routes the method call to an application code generic proxy that is an image of a local object [*Admitted prior art*, 0003: "Proxies may be employed...in non-distributed systems (where the proxy is a local image of a local object)], the method call interceptor accessible to application code to at least one of adapt or extend functionalities of the system [abstract : "input parameters might be filtered and changed by the user code" | column 8 «lines 8-11»]; and

the application code generic proxy that receives an intercepted method call, invokes a method on the object without crossing a remoting boundary [*Admitted prior art*, 0003: a proxy that is a local image of a local object does not have to cross a remoting boundary because the local object is not across the remoting boundary], receives results from the object and passes results to the entity that generated the intercepted method call based at least in part on the intercepted method call operability of the application code generic proxy modified by the application code, the application code generic proxy performs proxy pre-processing that includes custom, user-directed application processing to at least one of monitor or control the processing of a message between the application code generic proxy and the object [*Hollander*, column 4 «lines 25-47»: Hollander discusses diverting method calls to user-supplied code which can control application program execution flow and monitor application program activities | column 11 «lines 52-56»] that includes controlling at least one of amount of data sent to the object, a type of data sent to the object or which objects are invoked, in a manner to reduce response time [column 9 «lines 7-15»: the proxy controls the amount of data sent to the object by first determining whether the data is cached locally at the proxy. If it is, then the request is not sent to the object and is retrieved locally].

Arnold however does not expressly disclose a method call interceptor accessible to application code to at least one of adapt or extend functionalities. Arnold also does not disclose that the proxy is an image of a local object or that the proxy invokes a method of the object without crossing a remoting boundary.

With respect to an application code accessible method call interceptor and routing

intercepted methods to proxies, Hollander discloses these features. Hollander is directed to a method of intercepting function calls that originate from the operating system and routing them to a proxy. Hollander's user-supplied custom code corresponds to Applicant's claimed application code and Hollander's function calls read on Applicant's method calls. Hollander user-supplied custom code has access to a data structure that is populated by information, such as the call's parameters, relating to the intercepted call.

In this manner, the user-supplied code can perform pre-processing on the intercepted call by affecting its parameters and by way of this improved interception functionality, Hollander discloses that the user code has enhanced and extended capability to control, manage and handle system events [column 4 «lines 40-47»]. Hollander also discloses the optimization of the remote method call includes at least one of determining an amount of data sent to the object, determining a type of data sent to the object or determining which objects are invoked [column 11 «line 60» to column 12 «line 5» : Hollander teaches determining the type of parameters being passed to the object].

Since such functionality is well known and utilized in conventional systems, it would have been obvious to one of ordinary skill in the art to have modified Arnold's proxy system to included Hollander's interception and optimization functionality. One would have been motivated to enable an interceptor that was accessible to user-code so that users would have more flexibility to manage the object environment by providing extended capability to control and manage system related function calls.

With respect to the proxy limitations, these features were well known in the art at the time of Applicant's invention as described by Applicant's admitted prior art. Specifically,

Applicant admits that proxies that are images of local objects were well known in the art. These local objects are not located across the remoting boundary. Thus, any method invoked on these objects will not cross the remoting boundary.

It would have been obvious to one of ordinary skill in the art to have modified Arnold's proxy system to include the well known features as described in Applicant's specification. Such a modification to Arnold's system is an example of applying a known technique (admitted art of creating a proxy that is an image of a local object and invoking a method without crossing a remoting boundary on a local object) to a known system (Arnold's proxy system) ready for improvement to yield predictable results (improvement of Arnold's proxy system by increasing the functionality and capabilities of his proxy in a well known manner). *See MPEP § 2143*.

8. As to claim 2, Arnold discloses the object is located across a remote boundary [Figure 1 «item 606»].

9. As to claim 3, Arnold discloses the object is marshaled by reference [column 8 «lines 38-45»].

10. As to claim 4, Arnold discloses the object is marshaled by value [column 9 «lines 10-15»].

11. As to claim 5, Arnold discloses populating a call information data store with information associated with the intercepted method call, the call information data store is accessible to the application code generic proxy [column 9 «lines 16-27» | see also Hollander, abstract].

12. As to claim 6, Arnold discloses the call information data store is populated with at least one of: a method name and a class/interface defining method data [column 7 «lines 46-49» | column 9 «lines 59-66»].

13. As to claim 7, Arnold discloses the call information data store is a message object that be serialized and passed across a remote boundary [column 9 «lines 16-27»].

14. As to claim 8, Arnold discloses transferring control to a method in the application code generic proxy, the method in the application code generic proxy overrides a base class method defined in a base class object from which the application code generic proxy inherits [column 10 «lines 20-31»].

15. As to claim 10, Arnold discloses proxy preprocessing further comprises at least one of: transaction processing, object migration, monitoring remote method calls, caching local data, caching remote data, or controlling remote method call invocations [column 9 «lines 4-27»].

16. As to claim 11, Arnold discloses the application code generic proxy performing proxy post-processing after receiving the results from the object [Figure 7 «item 712»].



17. As to claim 12, Arnold discloses the proxy post-processing comprises at least one of transaction processing, monitoring remote method calls, caching local data, or controlling remote method call invocations [column 9 «lines 4-37»].

18. As to claim 13, Arnold discloses the proxy invoking the method on the object by invoking a method available in remote infrastructure [column 10 «lines 20-32»].

19. As to claim 14, as it does not teach or further define over previously claimed limitations, it is rejected for at least the same reasons set forth for claim 1.

20. Claims 15 and 20-22 are rejected under 35 U.S.C §103(a) as being unpatentable over Colyer, U.S Patent No. 5.903.725, in view of Clarke, in further view of Hollander, in further view of Applicant's admitted prior art.

21. All citations are to Colyer unless otherwise noted.

22. As to claim 15, Colyer discloses a method for interacting with an object, the method comprising:

employing at least one processor to execute computer executable instructions stored on at least one computer readable medium to perform the following acts:

creating a base class proxy object [column 7 «lines 37-51» : parent class];

creating an application code generic proxy, the application code generic proxy inherits from the base class proxy object [column 7 «lines 37-51» | column 11 «lines 45-64»];

overriding a base class method defined in the base class, the overridden method receives an intercepted method call [column 3 «lines 1-11» | column 12 «lines 4-18»];

intercepting a method call on the object [column 3 «lines 1-11»];

routing the method call to the application code generic proxy [column 3 «line 61» to column 4 «line 44»] without traversing a remoting boundary, wherein the application code generic proxy is an image of a local object [*Admitted art*, 0003];

adapting the application code generic proxy functionality based at least in part on the method call, the application code generic proxy performs custom user-directed proxy pre-processing [*Hollander*, column 4 «lines 33-47»] comprising transaction processing and machine learning to control at least one of an amount of data sent to the object, a type of data sent to the object, or which objects are invoked [*Clarke*, 0029: controlling the amount of requests that are sent to the object by determining whether or not to forward the request];

invoking the method on the object [column 3 «line 53» to column 4 «line 25»] based in part on the pre-processing;

receiving a first result from the object [column 3 «lines 44-59»]; and

returning a second result to the entity that generated the intercepted method call [column 3 «line 61» to column 4 «line 6»].

Colyer does not expressly disclose adapting the proxy functionality with the proxy performing pre-processing comprising transaction processing and machine learning nor does Colyer teach that the interception is made accessible to a developer to at least one of adapt or

extend functionalities. Colyer also does not disclose optimizing the remote method call by determining an amount of data sent to the object, determining a type of data sent to the object, or determining which objects are invoked. Finally, Colyer also does not disclose that the proxy is an image of a local object or that the proxy invokes a method of the object without crossing a remoting boundary.

With respect to the pre-processing functionality, Clarke discloses this feature. Clarke is directed to a client-server system with a proxy system in between [Figure 1]. Clarke's proxy is adaptive in the sense that the proxy utilizes machine learning in the step of preprocessing method calls from the client in order to optimize the invocation of the calls including achieving an optimal response time by controlling the amount of data sent to the objects [0029 | 0027 where : Clarke discloses that the proxy is adaptive in selecting appropriate servers with the proxy "learning over time which origin servers are most prone to overload"].

It would have been obvious to one of ordinary skill in the art to incorporate Clarke's teachings of an adaptive proxy into Colyer's system. Clarke discloses that an adaptive proxy helps control network congestion over the network. Thus, one would have been motivated to modify Colyer's proxy to be adaptive to optimize network efficiency of handling requests over the network.

With respect to an application code accessible method call interceptor and routing intercepted methods to proxies, Hollander discloses these features. Hollander is directed to a method of intercepting function calls that originate from the operating system and routing them to a proxy. Hollander's user-supplied custom code corresponds to Applicant's claimed application code and Hollander's function calls read on Applicant's method calls. Hollander user-

supplied custom code has access to a data structure that is populated by information, such as the call's parameters, relating to the intercepted call [abstract : "input parameters might be filtered and changed by the user code" | column 8 «lines 8-11»]. By way of this improved interception functionality, Hollander discloses that the user code has enhanced and extended capability to control, manage and handle system events. Hollander also discloses the optimization of the remote method call includes at least one of determining an amount of data sent to the object, determining a type of data sent to the object or determining which objects are invoked [column 11 «line 60» to column 12 «line 5» : Hollander teaches determining the type of parameters being passed to the object]. Hollander also discloses performing custom user-directed application processing to at least one of monitor or control the processing of a message between the application code generic proxy and the object [column 2 «lines 45-52»].

Since such functionality is well known and utilized in conventional systems, it would have been obvious to one of ordinary skill in the art to have modified Colyer's proxy system to included Hollander's interception and optimization functionality. One would have been motivated to enable an interceptor that was accessible to user-code so that users would have more flexibility to manage the object environment.

With respect to the proxy limitations, these features were well known in the art at the time of Applicant's invention as described by Applicant's admitted prior art. Specifically, Applicant admits that proxies that are images of local objects were well known in the art. These local objects are not located across the remoting boundary. Thus, any method invoked on these objects will not cross the remoting boundary.

It would have been obvious to one of ordinary skill in the art to have modified Colyer's proxy system to include the well known features as described in Applicant's specification. Such a modification to Colyer's system is an example of applying a known technique (admitted art of creating a proxy that is an image of a local object and invoking a method without crossing a remoting boundary on a local object) to a known system (Colyer's proxy system) ready for improvement to yield predictable results (improvement of Colyer's proxy system by increasing the functionality and capabilities of his proxy in a well known manner). *See MPEP § 2143.*

23. As to claims 20-22, Colyer discloses the object is located across a remote boundary [Figure 4], the object is marshaled by reference [column 11 «lines 51-54»] and the object is marshaled by value [column 4 «lines 9-12»].

24. Claims 17-19, 23 and 27 are rejected under 35 U.S.C §103(a) as being unpatentable over Colyer, Clarke, and Hollander, in further view of Arnold.

25. As to claim 17, Colyer does not disclose the preprocessing including load balancing, object migration, object persisting, monitoring remote method calls. In the same field of invention, Arnold discloses a proxy performing preprocessing including object migration, monitoring remote method calls, caching local data, caching remote data, or controlling remote method call invocations [column 9 «lines 4-27»].

It would have been obvious to incorporate Arnold's preprocessing steps into

Colyer's proxy object. Arnold discloses that such steps improve network response to method call invocation. Thus, one would have been motivated to combine the references to improve upon Colyer's proxy for the reasons stated in Arnold.

26. As to claims 18 and 19, Colyer does not expressly disclose performing post-processing. Arnold discloses the application code generic proxy performing proxy post-processing after receiving the results from the object [Figure 7 «item 712»], whereby the post processing includes transaction processing, monitoring remote method calls, caching local data, or controlling remote method call invocations [column 9 «lines 4-37»]. It would have been obvious to one of ordinary skill in the art to incorporate Arnold's post processing steps into Colyer's system to enable caching of objects which improves network response to method call invocation.

27. As to claims 23 and 27, as they do not teach or further define over the previously claimed limitations, they are rejected for at least the same reasons set forth for claims 15 and 17-19.

### ***Conclusion***

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period

will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to DOHM CHANKONG whose telephone number is (571)272-3942. The examiner can normally be reached on Monday-Friday [8:30 AM to 4:30 PM].

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John Follansbee can be reached on 571.272.3964. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Dohm Chankong/  
Primary Examiner, Art Unit 2452